TMOD Technology and Science Program News
June, 2000. Issue 12.

Mars Rover Mission Distributed Operations

Paul Backes and Jeffrey Norris
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California

Introduction

Internet-based rover mission operations will enable participation in planetary rover missions by an increased number of scientists and at reduced costs. The Web Interface for Telescience (WITS) and the Multimission Encrypted Communication System (MECS) have been developed to enable Internet-based Mars lander and rover mission operations. WITS provides downlink data visualization and sequence generation and MECS provides secure Internet-based communication. WITS and MECS were used by the Mars Polar Lander (MPL) mission for Robotic Arm sequence generation during operations readiness tests (ORTs) and during the post-mission field test. This article describes WITS and MECS and explains how they were utilized in the Field Integrated Design and Operations (FIDO) rover desert field test of May 7-18, 2000. Further information on WITS and MECS, as well as a public-outreach version that can be downloaded and run on a PC, can be found at the URL http://robotics.jpl.nasa.gov/tasks/wits/.

Internet-based operations provide various benefits. Use of the public Internet for distribution of mission data eliminates the costs of dedicated leased lines which have been previously used to support remote users. The MECS system makes it possible to use the Internet for data distribution by providing the necessary security and data encryption. Enabling users to participate in a mission from any location on the World Wide Web reduces both housing and operations center infrastructure costs. Expenses can be further reduced by enabling scientists to continue with other work at their home institutions during the mission. This is particularly important for long-duration missions and those which continue into extended mission phases. Internet-based operations are valuable for engineers and scientists at JPL; they can support multiple missions from one location, e.g., their office and utilize computing resources outside of the mission operations area. Pre-mission ORTs can also benefit from Internet-based operations, because participants from other institutions can fully participate in ORTs without the additional cost (both in time and money) of having to travel to JPL for each test. Thus, more tests can be performed which involve all mission participants.

The FIDO rover is a prototype of the Athena rover that is being considered for the 2003 NASA Mars mission. FIDO is equipped with many of the elements of the Athena payload. It is used to simulate the complex surface operations expected of the Athena Payload, focusing on identification of rock targets, approaching the targets and conducting in-situ measurements, and drilling and verification of cores. Stereo cameras on the rover's mast and body are used to image the surrounding terrain. An instrument arm places instruments on selected surface targets. A coring drill is used to extract rock samples.

The FIDO field test was used to evaluate new rover operations scenarios for the potential Athena rover mission while utilizing Internet based operations enabled with WITS and MECS. Rover operations are conducted independently of a lander and the rover can traverse long distances from the landing site. During the field test, operators used WITS and MECS to visualize downlink data and generate command sequences from JPL; Ithaca, New York; Birmingham, Alabama; Flagstaff, Arizona; St. Louis, Missouri; and Copenhagen, Denmark. In the operations readiness tests leading up to the field test, scientists Steve Squyres and Ray Arvidson and their colleagues from Cornell University and Washington University, respectively, were able to lead the tests from their remote locations. The field test demonstrated that WITS and MECS will enable efficient distributed operations in the next Mars rover mission.

System Architecture

WITS and MECS are designed to support multiple lander and rover missions. The operations architectures for their use in the MPL mission and FIDO rover desert field test are shown in Figures 1 and 2.

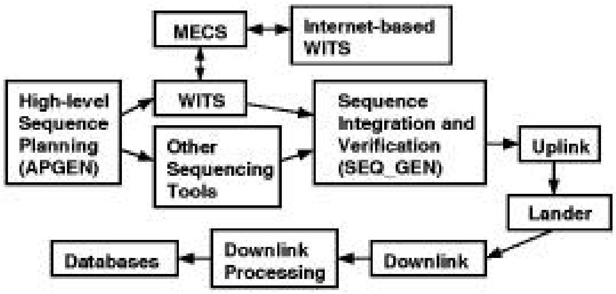

Figure 1: MPL Mission Operations Architecture

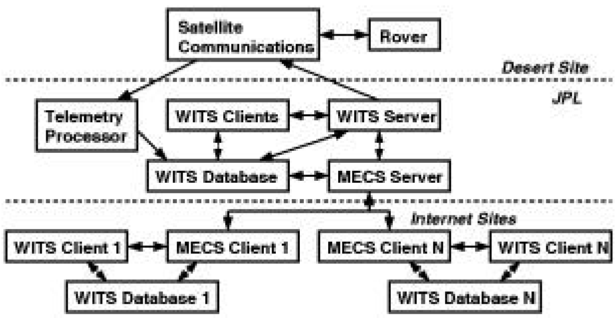

Figure 2: FIDO Operations Architecture

For FIDO operations, WITS is used by itself. In the MPL mission, the APGEN (Activity Plan Generator) sequencing tool generated high-level sequences. WITS added the low-level commands to satisfy the high-level goals within resource allocations for the Robotic Arm and Robotic Arm Camera. SEQ_GEN (Sequence Generator) was used to integrate all of the low-level sequences (including those for other instruments) into the final uplink sequence.

WITS includes a database, server, and multiple clients. The database is a structured file system that holds downlink data products and uplink sequence information. The server provides communication between clients and some database operations. Clients are distributed over the Internet and provide the interface needed to view downlink data and generate command sequences. Other tools, e.g., for planetary ephemeris or sequencing, can interact with WITS by reading and writing to the database or by direct communication with the server. WITS is implemented using the Java2 platform, including the Java3D and Java Cryptography extensions.

Internet Security

To enable collaboration in daily sequence generation by Internet-based scientists, a secure and efficient means to transfer data is needed. MECS was created to provide the required secure Internet-based communication. MECS was integrated with WITS for MPL and FIDO, and is designed to work with other mission tools as well. MECS operates in a fashion that is transparent to the remote user: Files simply appear as they become available and connections are made securely without any additional effort on the part of the user. MECS provides two types of encrypted communication for WITS: 1) automated delivery of database updates to Internet-based clients, and 2) encrypted communication between the clients and server. Since encryption was not required for the FIDO rover field test, the communication features of MECS were used without encryption.

MECS connections are authenticated using the NASA Public Key Infrastructure (PKI). After authentication, communications are made through SSL (Secure Sockets Layer) and are encrypted using the Triple-DES-EDE3 algorithm. Some of the advantages of MECS for Internet-based data distribution are: 1) MECS does not require remote users to request a data update; data is automatically delivered as it becomes available; 2) The MECS administrator can specify on a user-by-user basis exactly who will receive a particular type of file or directory; 3) Since MECS is written entirely in Java, it can run on most computers without modification. 4) MECS provides a high level of data security by using the SSL algorithm to perform authentication, and the Triple-DES-EDE3 algorithm for encryption. 5) MECS clients can be allowed to transfer files back to the mission control center. Files from users are stored on the server in a compressed, enveloped form that allows them to be scanned for hostile code. Since every client is authenticated using the NASA PKI, unauthorized users cannot transmit files to the server.

MECS is implemented as two Java programs, server and client, using the publicly available Entrust Java Toolkit's low-level authentication and encryption capabilities. For each mission, there is typically one server, operating behind a mission firewall, and many clients, one on each remote user's machine.

Figure 3 illustrates the steps necessary for a single MECS transaction. Steps 1 and 2 occur once, before the beginning of the mission, while steps 3 through 9 occur for every transaction. In step 1, a remote user must obtain a security profile from NASA PKI, which requires appearing in person at a NASA center security office. A security profile is a set of files on a floppy disk, protected by a password, that contain a user's private key. Users need their private keys

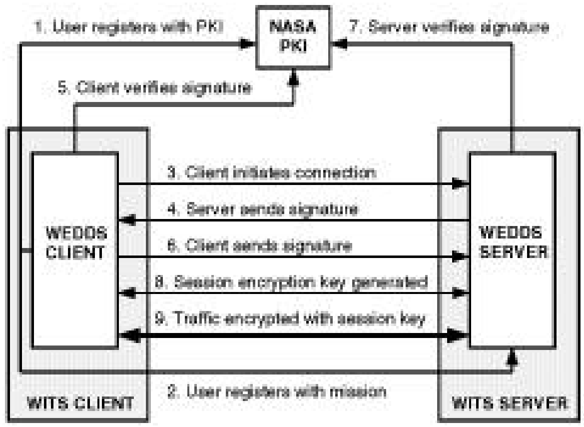

Figure 3: MECS Secure Communication Steps

to positively identify themselves online. The WITS server is also issued a security profile so that it can prove its identity to remote users. In step 2, each user must contact the mission administrator and request that their profile be given access to mission data.

Steps 3 through 9 are repeated for every transmission from the client to the server or from the server to the client. In steps 4 through 7, the server and client exchange digital "signatures" generated from their security profile. They verify these signatures by communicating with the NASA PKI. This process, known as SSL authentication, relies on the fact that it is nearly impossible for someone to generate another user's digital signature without that user's private key and password. In step 8, the last step in establishing an SSL connection, the client and server use the Diffie-Hellman key agreement protocol to generate a unique symmetric encryption key that will be used to encrypt the traffic for this transaction. This encryption key is never sent as clear-text over the Internet, and all subsequent traffic for this transaction is encrypted, making MECS transactions invulnerable to eavesdropping or "packet-sniffing" attacks. In addition, every MECS transaction is protected by a new encryption key.

Downlink Data Visualization

Downlink data from the lander or rover is provided by WITS by means of various 'views'. Panorama, Overhead, Wedge, and 3D views with data from the FIDO rover desert field test are shown in Figure 4. The Results Tree window enables the user to select views to be opened based upon rover locations. The Plan window provides available views based upon a specific uplink plan. A plan includes view definitions and sequence information for generating one uplink sequence.

The Descent view provides images taken from orbit or during descent, and shows the landing location. The Overhead view shows the immediate area around the rover from above using various data formats (e.g., texture map, elevation map, contour map). The Panorama view is a mosaic of images taken by a stereo camera located on the deployable mast of the FIDO rover. The Wedge view displays one image of the panorama with various viewing options. The 3D view provides a 3D solid-model visualization of the rover and terrain. Animated sequence simulation and state are visualized in the 3D view. The Contrast Adjuster view enables the pixel intensity range to be adjusted for a Wedge view image.

Selecting a point in an image causes the x,y,z position at that point on the surface to be displayed. A target or rover waypoint can be created at the selected point via a menu option. Targets and waypoints are named 3D locations that can be used as parameters in sequence macros. Waypoints are generally used as locations for the rover to traverse through and targets are locations for science activities such as drilling.

Sequence Generation

WITS provides various windows and features for command sequence generation. The Sequence window is used to generate a command sequence, which has a hierarchy of elements: Sequence, Waypoint, Request, Macro, and Step. A Request represents a high-level task. A Macro is the WITS element used to specify commands and parameters. Macros have expansions into Steps. A Step is a low-level command that will be uplinked to the spacecraft. WITS can generate output sequences in various formats, e.g., Spacecraft Activity Sequence File (SASF) format for the MPL mission and a FIDO-specific format for the FIDO rover.

The left side of the Sequence window displays a list of macros that can be inserted into a sequence. Multiple lists of macros are available. A Macro window is used to specify the parameters for a macro. A macro-specific algorithm expands the macro into one or more command steps. A macro can generate view objects that are displayed in the views to indicate what actions the macro is producing, e.g., footprints on the terrain for imaging commands. A valuable feature of the Sequence window is sequence state visualization. When a step in the sequence is selected, then the rover state is updated in the various views with the state

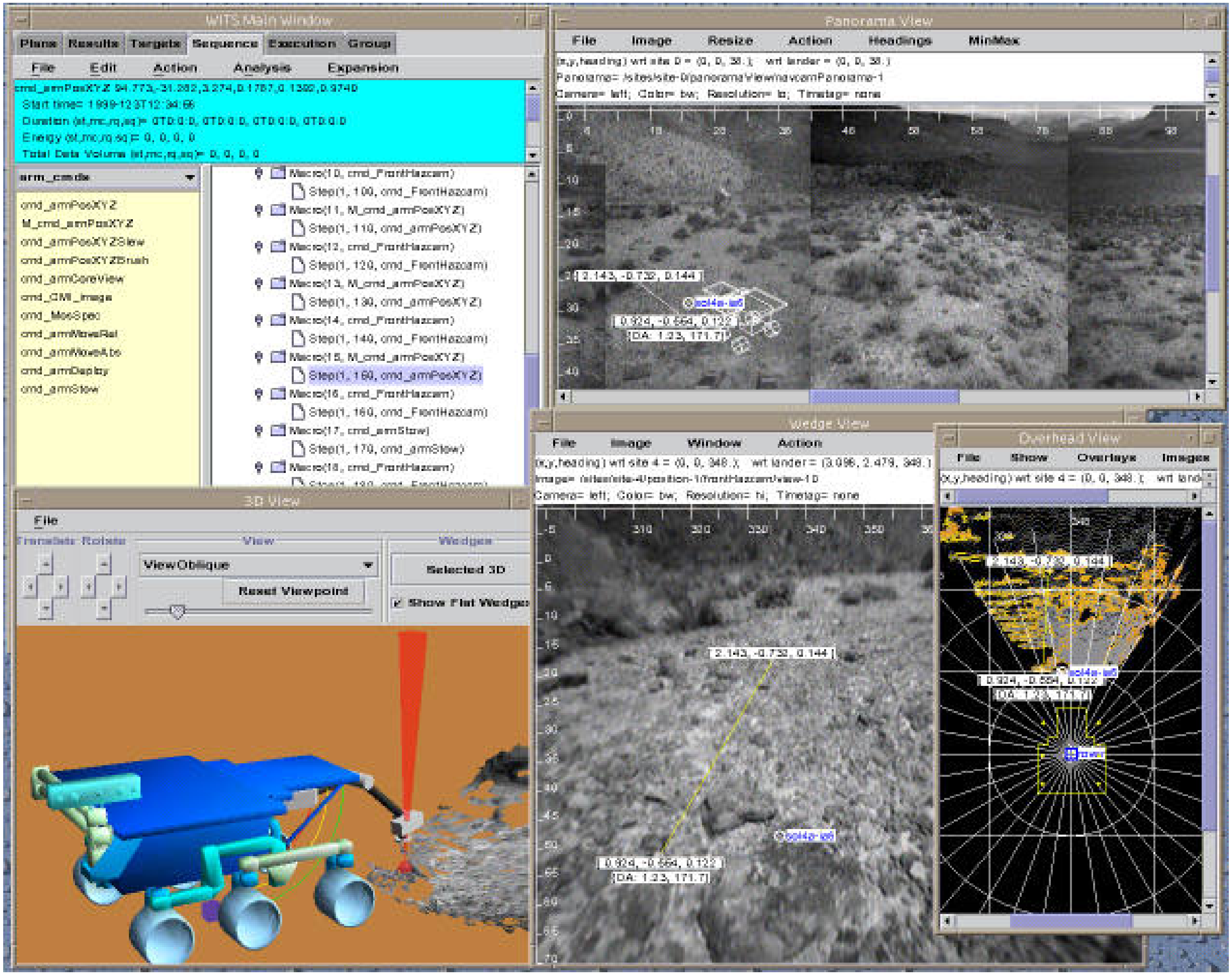

Figure 4: WITS Visualization and Sequence Generation

computed at the end of the step.

Resource analysis and rules checking are provided for sequence generation. The duration, energy, and data volume for each step of the sequence are computed and stored along with the cumulative duration, energy, and data volume at each step. Rules checking ensures that a sequence is valid relative to specified sequence rules.

Automated sequence report generation is provided. The sequence report is an HTML document that is produced with detailed information about a sequence. The report includes a description of the rover state at the start of the sequence and at each step of the sequence, screenshots of views at steps of the sequence, a resource report, the output sequence, and one movie for each view showing the rover motion through the sequence.

Distributed Collaboration

WITS provides features to support efficient collaboration within a distributed operations team. Users can input targets and command sequences and save them to the common server, as well as load the targets and sequences from the other users. A group mode is provided. If a user enters group mode, then the user can send and receive group commands. Group commands, which are executed on each group member's WITS client, include: 1) open view, which opens the specified view; 2) add marker, which displays a 3D marker with the originating user's name in all views; and 3) white-board commands that send messages to either one specific group member or the whole group. An announcement feature pops up a window with a message in all group users clients. When a user joins the group, the state of the group, including all views and markers, is automatically initialized.

Acknowledgements

The work described in this paper was funded by the TMOD Technology Program, the Mars Exploration Technology Program, the SBIR Program, and the Mars Polar Lander Mission. The FIDO rover field test was led by Steve Squyres of Cornell University, Ray Arvidson of Washington University, and Paul Schenker, JPL task manager for FIDO rover development.